

Layered Context Modeling in Situated Information Processing

Augusto Celentano

Dipartimento di Informatica, Università Ca' Foscari Venezia
auce@dsi.unive.it

Abstract

In this paper I propose a model for situated information processing based on a set of layers, each related to a different processing phase and to a different context type. The model is illustrated with reference to a system for processing semantically rich documents, typical of business and juridical domains. Starting from the conceptual level of document definition, through operations of selection, specialization, instantiation and presentation document are adapted to the situation to which they apply and to the local context in which they are issued.

1 Introduction

The term *situated computing* parallels the term *context-aware computing* and is progressively replacing it in the literature on mobile and contextually informed system design [10]. Situated computing draws from situated cognition [8] "which holds that where are you and when are you there matters, and that the state you are in affects what you do" [12].

The word *situation* is used together with or instead of the word *context* to denote the environment affecting the way a user executes a task, interacts with a device, accesses information, and so on. The concepts subsumed by the two words present differences (sometimes subtle) in different authors. In most cases the situation is a component of the user context, referred to personal traits rather than to the surrounding environment, to the equipment, to the available resources, etc. The situation describes the user with respect to the environment, while the context traditionally describes the environment with respect to the user.

I would suggest a different view on situation and context. According to authoritative dictionaries, a *situation* is the status of a part of the world with respect to conditions and circumstances; it defines the relations between something and its surrounding [15]. In my perspective the situation extends the notion of context, which remains primarily attributed to physical, observable features of the user envi-

ronment. The situation tries to capture aspects of the user environment at a higher level including primarily the user goals, plans, activity and history [13]. As such, situation is not related to the immediate and evolving status of the user, but reflects his/her attitudes and motivations with respect to a wider plan of actions related to a goal.

Keeping distinct the two concepts naturally leads to a layered model of context definition and analysis. The model I propose is focused on data intensive applications, where information goes through several processing stages including operations like selection, specialization, instantiation and formatting of its content. I shall take as a working case study the processing of documents characterized by rich semantic content, defined by formal regulations, having important social effects on the users. Such documents are contracts, agreements, acts, notary deeds, in general documents with a juridical relevance, whose contents change as a function of the situation they address. Each final document serves a well defined case within the context of the user and of the surrounding normative world.

The discussion could as well be focused on the domains of data intensive web sites, e-commerce applications, e-government portals, etc.. As juridical documents, such applications are built according to social and juridical rules, and must adapt to the user's goal, profile and environment. However, dealing with prose text rather than with records and tables might better express the requirements of complex and flexible adaptation at several levels, including issues related to the correct linguistic expression, to domain related jargon and to the writing style.

The paper is organized as follows: after briefly reviewing some literature on situation, context and adaptation in Section 2, Section 3 presents a reference framework for adaptable documents definition and processing. Section 4 introduces the concept of layered context awareness, and Section 5 defines four levels of situated information processing related to different context layers. Section 6 discusses issues related to situation and mobility, and draws the concluding remarks.

2 Situation, context and adaptation

Situated cognition, like context aware computing, builds on the adaptability of information content, applications and interaction through the formalization of how a situation modifies the perception a user has of the environment, hence the way information processing and interaction systems work.

The notion of context has been widely extended since the initial focus on location; the definition by Dey [9] is still taken as authoritative, but many authors try to make more explicitly the presence of components related to the user “per se” (see for example [14, 17]). Extending the analysis of the user situation to include the user plans and goals in executing tasks and accessing information requires a view of the context also in cognitive and linguistic domains [18]. A complete view of context is presented by Bradley and Dunlop [4] who explore a multidisciplinary approach addressing in a unified view different domains.

The notion of context as a description of the situation of a user executing an application has greatly influenced the adaptation of information and documents, evolving from ad hoc processing up to automated systems for generating and tailoring content and presentation to the user environment and needs. Adaptation occurs at several levels: media and layout [22], logical structure [3], inter-media relations [5], and cognitive impact [11]. Consequently, context modeling has been extended from the simplest model based on unstructured attribute/value pairs up to models based on logic and ontologies [7, 20, 19].

Context adaptation of structured information in the area of personal databases has been investigated by the Context-ADDICT (Context-Aware Data Design, Integration, Customization and Tailoring) project¹. The project aims at supporting mobile users through the dynamic hooking and integration of new information sources, delivering only appropriate, context-based portion of data to their mobile device. Adaptation is based on an ontological representation of the application domain and of data source contents [1, 2, 21].

3 Semantic-based document processing

In scenarios bound to juridical and normative domains the way a document is drawn depends on precise rules that enforce a change or set a status in the world concerning the document issuer, the addressee and the social relations around them. In a broad sense, such documents are the results of the execution of a procedure which follows a plan defined by a set of regulations adapting to variants depending on the situation raised by the procedure actors.

The two terms *situation*, introduced in Section 1, and *case* denote two related but distinct concepts. From an in-

formation systems point of view, the concept of situation is closer to the idea of class than to the idea of instance. For example, a situation is the set of features and circumstances which distinguish a purchase of an apartment in a condominium within the class of real-estate transactions; the situation describes the rules, the clauses and the constraints that apply specifically to apartments and not, e.g., to houses or to industrial buildings.

A *case* is a specific occurrence of a situation, in which all the information about the subjects and objects involved is instantiated. Completing the example above, a case is the purchase of such and such apartment, with such and such sellers and buyers, according to such and such financial conditions.

The gap between the two concepts is large, because situations can be more or less detailed, thus representing a continuous range of information types and instances up to a point in which all information is instantiated. Somewhere in this range there is a border, which depends on the domain and is subject to some degree of discretionality.

The paper [6] introduces the concept of *situated document* to denote documents whose processing is defined by the user situation, which does not include only physical or technical context parameters, but also (and mainly) the role of the document, the effects of the document on the user and on his/her social relations, and the user goal, sharing many features of the scenario of this work. I shall take from the architecture for situated documents only the basic ideas described hereinafter, which characterize a broad class of semantic-based adaptable documents, and information containers in general.

Document collection. Documents are instances of classes organized in collections (called *suites* in [6]) grouping related documents. A collection is bound to an application domain targeted to a specified goal. It defines both the goal and the set of documents needed to pursue it. An example of a document collection is the set of documents needed to buy an apartment in a condominium. According to the Italian law, the collection includes a preliminary contract (usually a private agreement), a notary deed of purchase, and documents for the real-estate registry office, the tax offices, the municipality archives, etc.. Other documents, such as a mortgage deed, a declaration of marital status of the buyer, a procurement act, and so on, might be required according to the *situation* in which the apartment is sold, related to the seller, the buyer, the apartment status, and a set of condition set by the local administration which may be different in different locations.

Conceptual document. Each document of a collection is a *conceptual document*, a model for a class of documents whose overall structure, goal and meaning is defined, but where specific data, dependent on the actual case, are not

¹<http://poseidon.elet.polimi.it/ca/>

defined. For example, a deed of purchase has an almost invariant structure in which actual data of deed subjects and objects can be embedded. It may have different *clauses*, each corresponding to a different status of the apartment (e.g., there is a mortgage on it), of the seller (e.g., is a minor), of the buyer (e.g., is represented by a proxy), or of the location (e.g., the building in which the apartment is of historical interest, and subject to a bond).

Concrete document. A conceptual document collection is instantiated into a set of concrete documents through a sequence of processing phases pertaining to an adaptation plan that develops in several layers.

Adaptation to the situation concerns primarily the choice of the documents of the suites to be instantiated; this phase is driven by the situation at its most general level. Then, for each selected document type, the adaptation concerns the selection of the pertinent variable clauses which complete the invariant content with details about the specific situation to be described. The outcome of these two phases is a set of document templates which can be used for a class of cases differing only in the instances of the subjects and objects involved, not in the situation they apply.

A semantic-based document model must include at least three elements: (1) a structure, almost invariant, which defines the document main role and the type of content, as required by the class of situations in which it is used; (2) a set of variable and possibly optional components, called *clauses*, which depend on a subset of the information describing the situation; and (3) a set of data instances describing the actual case.

Up to now I have not considered the document final presentation. I have introduced selection and adaptation of the document content, without considering how the content appears. Nevertheless, the presentation and its variants are one of the relevant outcomes of the part of the context that describes the environment, the devices, the user preferences, i.e., the elements that do not influence the document content and meaning, but help to convey it in the optimal way.

This element enters as the final phase of adaptation in the process described above, and is represented by a further component of the document model: a set of rules which define how to instantiate the physical form of the document, according to a set of parameters which describe the related part of user context.

4 Situation and layered context-awareness

The instantiation/adaptation process described in previous section relies on four types of information which enter in four stages of document processing: (1) the external situation and the user goal; (2) the information about facts specializing the situation to a narrow case class; (3) the details

about the data instances of the specific case; (4) the environmental context defining the document physical aspect.

To characterize the four information types, we need to define what we mean with *context* in a more precise way.

As recalled in Section 2, there is still some vagueness about what the word *context* means. There is a general agreement about associating it to the environment surrounding the execution of an application, or the processing of an information. Anyway, what is context as opposed to application data is still matter of discussion.

The scenario described in this paper is a good example of this debate. Since early '80s the automatic generation of documents in some well defined domains has been a reality. The business domain and the juridical domain are the two areas which have gained benefit from such automation. Systems for drawing contracts and notary deeds have been designed and are widely used since then. At that time, the word context had no special meaning, and almost all the office document processing applications used all the data at the same level. A database holding all the relevant data was consulted to extract different types of information, such as: switches to select the proper template, coherent with the set of data instances; flags signaling the need to include or exclude clauses; actual data instances to fill out the blanks in document templates; rules to verify the coherence of the different document clauses [16]. Often, also information about the final presentation, such as the document format and the type of printing has to be known during all the process. A few better structured application programs used to differentiate at least the selection of the document template and the final format from the data needed to instantiate the document.

Since then, the (once) new concept of context, new information structuring methodologies and new models for knowledge representation have modified the scenario, and we are now able to distinguish, practically if not theoretically, the different types of data occurring in a complex information processing application, and to arrange them in layers, each layer having a specific goal with respect to the final result.

In processing a document we proceed through a series of phases, in each phase selecting, modifying, adapting or instantiating some components of a comprehensive document model including document types, structures, templates and sentences in natural language. A set of rules supplement the model defining, in some way, constraints and mutual relationships. A simple example, solved at different levels of smartness in old automatic processing systems, is the linguistic concordance among the grammatical properties of the text and the variable data to be embedded. According to the juridical style of language used in notary deeds, which is very precise, a sentence stating that a (set of) purchaser(s) buys (buy) an apartment by a (set of) seller(s) has four lin-

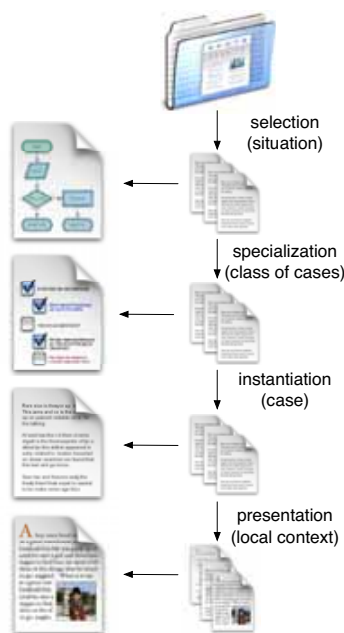


Figure 1. Layered document processing

guistic variants according to the cardinality of each party in the deed, and more complex situations can be raised by gender biased expressions.

The processing phases, and the related ruling descriptions, are not at the same semantic level. Each phase uses a set of data for a specific goal, and produces a specific type of transformation. Some data are used only in one level, other data are used, with different roles, in different levels.

I denote this situation with the term *layered context-awareness*: at each layer a context environment is defined, which is used to modify the way an information is generated, processed or used. The way the context is modeled and described in concrete may depend on the layer. While admitting that the choice of a suitable model is the crucial point of this approach, in this paper I shall overview the pros and cons of some models, leaving to future work the design of suitable and formally processable choice criteria.

5 Layered document processing

Document processing is done for each layer assuming three types of input: (1) a document structure (a type, a template), which is the source for the adaptation of that phase; (2) a context instance, suitable for that processing level; (3) a set of data representing the situation or case specific information for that processing layer. Figure 1 illustrates the four layers.

Layer 1: Selection. The uppermost layer of processing is

related to the choice of the documents in a suite; the choice depends on the situation in its widest meaning. It depends on the documents' role for fulfilling the user's expectation. The goal is linked to the application domain, to the initial state of the user world according to the domain, and to the final state to be reached. It could therefore be described with a reference to the semantics of the part of the world involved.

It is worth to note that in this layer the distinction between *context* and *application data* might blur more than in the other layers. If we think of automatic generation of notary deeds, it might be hard to say that the information about what deeds need to be generated is not part of the application input data.

Nevertheless, at this stage we could well ignore the *information* about the subjects and objects involved (which is known as the case), but we need to know which is their situation. We are reasoning at an abstract level about the identities (the instances), but at a concrete level about the circumstances. Hence, it's plausible to assume that most of information at this level is about context and not about content.

Layer 2: Specialization A second layer of processing is the identification, for each document selected at layer 1, of the relevant clauses. The situation is described in more specific terms because it is (also) based on actual data about the subjects and objects of the document. For example, while the presence of a mortgage is an explicit information that must be provided as part of the context in which the transaction is done, the presence of a minor among the parties, identified by the birth date, which is application data, requires he or she to be represented by a proxy, or the presence of witnesses to validate the deed (thus becoming an element of the situation).

Layer 3: Instantiation. The third layer concerns directly the composition of the document content. The document template, selected in the suite and adapted in its skeleton to receive the appropriate data, is filled with the variable data that describe the subjects and objects of the transaction. We assume that in this layer the final presentation is not yet taken into consideration, therefore it is most concerned with processing application data: the contribution of context is therefore very marginal. Nevertheless, some dependences from parameters not immediately recognizable in the application data, still not related to the physical aspect of the document, could be found at this level.

The more immediate (and maybe abused) example is the adaptation of the document language. Should such a change be reflected in the document template used? As such, it should be raised to the previous layer. Or should it correctly be considered as part of the instantiation, since the *meaning* of the template in the current case does not change, but only

its external appearance (even if it is not matter of simple formatting)? This case, in its simplicity, shows the type of ambiguity that even in a controlled architecture like the one proposed here affects the classification of context with respect to application data.

Another type of context, still more difficult to grasp, is the choice of the linguistic (or literary) style of the final document. As we are facing prose text and formally relevant documents, processing is not just a matter of filling a few placeholders in a template. Natural language processing techniques exist that may help to produce good text according to adaptable grammar and rhetoric rules. In this case, however, we risk to assign to the word *context* a different and broader meaning, involving the cultural environment of the users. The concept of adaptation broadens too; besides trespassing the scope and the goals of this paper, approaching such an issue requires a different focus and a different skill.

Layer 4: Presentation. The fourth layer includes the final instantiation, i.e., the adaptation of the document to the local context of the user, of the environment, of the device, according to the requirements, the guidelines and the techniques so deeply discussed in the literature about context-awareness. Therefore, I do not enter into this level, which is anyway important in resource critical scenarios such as those based on mobile technology.

The processing functions needed in the different layers demand different models for defining the situation and the context. In the first layer, selection, the situation definition must express a possible complex knowledge about the application domain procedures and about the role of information. It must be rich enough to express relationships between classes of facts, which can be implemented as relationships between sets. Models based on ontologies, despite their complexity, could serve the purpose; for efficiency reasons, this level could be categorized as a collection of *sub-situations*, each stable enough to be selected with simple mechanisms. Models based on logic could also be used profitably, mainly for expressing direct dependences between facts related to the situation and single conceptual documents.

In the second layer, specialization, the context denoting the situation is derived from a subset of data about the transaction, plus a set of information about how to execute the transaction. This set will not be used to adapt the document text, rather it will be used to select the proper clauses. A suitable model to describe the interrelationships between these two classes of information must be able to manage rules and constraints, therefore a logic based context model is to be preferred.

At the third layer, instantiation, much depends on what is considered context adaptation with respect to document

pre-selection, and to what extent a natural language based processing is considered necessary, as discussed above. At the most demanding side of the problem, however, I do not see context models so rich to account for such a complex processing, except that models and formalisms for knowledge representation could help to reason about context too.

The fourth layer, presentation, mimics the adaptation functions commonly assigned to context-aware systems, therefore a large fan of context models exists to support it.

6 Situated computing and mobility

In the current literature context awareness and situated computing are strongly related to mobility, and have received increasing attention from the growth of mobile information processing. Nevertheless, the concepts are disjoint. Mobility induces technology driven constraints, requires best use of resources both in terms of computing and communication power, and challenges the user cognitive effort. Mobility spontaneously leads application designers to consider context-awareness as a fundamental feature, since the information processed is subject to change during a session: information related to the location and to the environment, resources related to the communication and to the device, and the goal of the user may change as the situation of the user and the surrounding environment change.

Is situation-awareness a typical property of mobility? The answer is, of course, yes. Is situation-awareness meaningful only in presence of mobility? Despite the largest effort has been devoted to study situation-awareness in relation to a changing user environment, the answer is *no*. Situation expresses a concept broader than the environmental context defined in pervasive systems, and applies also when the environment does not change.

As mobile devices increase their capabilities, the difference with desktop devices will no longer be a matter of functions, but of opportunity. A wide and authoritative scientific literature foresees the use of personal mobile devices to carry out most of the activities today done with desktop computing devices: the management of medical records and medical tests in the healthcare domain, the electronic transactions in the business domain, on-site assistance in case of disaster in the emergency domain, and so on.

The information processed in such cases needs to be adapted to the user goals, before than being adapted to a device. What part of information about a patient is relevant for a doctor depends on what role the doctor is playing when at the patient home, in the hospital, in the operating room, in a mobile rescue unit.

The layered context architecture illustrated in Section 4 is only a first step that needs to be completed with inter-layer paths, which assure that information processed in a higher layer can be further processed in a lower layer. For

example, going back to the juridical domain used throughout this paper, the selection of a conceptual document within a document suite must be compatible with the possibility of processing the document down to the concrete instance. A professional user accessing a real-estate transaction document suite through a mobile device could receive all the information about a deed to be stipulated, but could not be able to produce the final document, e.g., because of limited access to public registry offices from a mobile device, a process which is managed at document instantiation time.

Such a requirements impacts on the models used for defining the context, and on the processing architecture that integrates context and application data processing, and requires to analyze at a deeper extent the ideas here proposed as a first step to approach the problem.

References

- [1] C. Bolchini and E. Quintarelli. Filtering mobile data by means of context: a methodology. In *Proc. OTM Workshops 2006 - Context-Aware Mobile Systems*, pages pp. 1986–1995, 2006.
- [2] C. Bolchini, F. A. Schreiber, and L. Tanca. A methodology for very small database design. *Information Systems*, 32(1):61–82, March 2007.
- [3] S. Boll, W. Klas, and J. Wandel. A Cross-Media Adaptation Strategy for Multimedia Presentations. In *ACM Multimedia*, pages 37–46, 1999.
- [4] N. A. Bradley and M. D. Dunlop. Toward a multidisciplinary model of context to support context-aware computing. *Human-Computer Interaction*, 20(4):403–446, 2005.
- [5] A. Celentano and O. Gaggi. Context-aware design of adaptable multimodal documents. *Multimedia Tools and Applications*, 29(1), 2006.
- [6] A. Celentano and F. Pittarello. Situated multimodal documents. In *MIMIC '07, 1st Int. Workshop on Management and Interaction with Multimodal Information Content*, Regensburg, Germany, September 2007.
- [7] H. Chen, F. Perich, T. Finin, and A. Joshi. SOUPA: Standard ontology for ubiquitous and pervasive applications. In *MobiQuitous 2004, 1st Annual International Conference on Mobile and Ubiquitous Systems, Networking and Services*, pages 258–267, Cambridge, MA, 2004.
- [8] W. J. Clancey. *Situated cognition: on human knowledge and computer representations*. Cambridge University Press, 1997.
- [9] A. Dey. Understanding and Using Context. *Personal Ubiquitous Computing*, 5(1):4–7, 2001.
- [10] P. Dourish. A foundational framework for situated computing. Position paper for the CHI 2000 Workshop on Situated Computing: A Research Agenda.
- [11] N. Elouazizi and Y. Bachvarova. On cognitive relevance in automatic multimodal systems. In *Proc. IEEE Conf. on Multimedia Software Engineering*, pages 418–426, Miami, FL, 2004.
- [12] J. S. Gero. Situated computing: A new paradigm for design computing. In A. Choutgrajank, E. Charoenslip, K. Keatruangkamala, and W. Nakapan, editors, *CAADRIA03*, pages 579–587, Bangkok, 2003.
- [13] K. P. Hewagamage and M. Hirakawa. Situated computing: A paradigm to enhance the mobile user's interaction. In S. K. Chang, editor, *Handbook of Software Engineering and Knowledge Engineering*. World Scientific, 2000.
- [14] A. Jameson. Modeling both the context and the user. *Personal Technologies*, 5(1):29–33, 2001.
- [15] Merriam-Webster. *Merriam-Webster's Collegiate Dictionary, 11th Edition*. Merriam-Webster, 2003.
- [16] P. Paolini, F. Garzotto, S. Mainetti, F. Barbic, A. Celentano, and F. Liguori. Knowledge based document generation. In W. Lamersdorff, editor, *Office Knowledge: representation, management and utilization*. North-Holland Publ. Co., 1988.
- [17] D. Petrelli, E. Not, M. Zancanaro, C. Strapparava, and O. Stock. Modelling and adapting to context. *Personal Ubiquitous Comput.*, 5(1):20–24, 2001.
- [18] A. Schmidt. A layered model for user context management with controlled aging and imperfection handling. In *MRC 2005, Modeling and Retrieval of Context*, volume 3946 of *LNCS*, pages 86–100. Springer Verlag, 2006.
- [19] T. Strang and C. Linnhoff-Popien. A context modeling survey. In *UbiComp, 1st International Workshop on Advanced Context Modelling, Reasoning and Management*, pages 34–41, September 2004.
- [20] T. Strang, C. Linnhoff-Popien, and K. Frank. CoOL: A context ontology language to enable contextual interoperability. In *DAIS 2003, Distributed Applications and Interoperable Systems, 4th IFIP WG6.1 International Conference*, volume 2893 of *LNCS*, pages 236–247, Paris, France, 2003. Springer Verlag.
- [21] L. Tanca. Context-based data tailoring for mobile users. In *Proc. BTW Workshops 2007*, page 282295, 2007.
- [22] J. van Ossenberg, J. Geurts, F. Cornelissen, L. Rutledge, and L. Hardman. Towards second and third generation web-based multimedia. In *Proc. Tenth International World Wide Web Conference*, pages 479–488, Hong Kong, 2001.